

Implementasi Pemetaan Robot Roda Mecanum Otonom Berbasis *LIDAR* dengan *SLAM*

Senanjung Prayoga^{1*}, Diki Sahidan^{1*}

*Politeknik Negeri Batam

Jurusan Teknik Elektro

Parkway Street, Batam Centre, Batam 29461, Indonesia

E-mail: senanjung@polibatam.ac.id, dikisahidan1@gmail.com

Abstrak

Artikel ini mengulas tentang autonomous mobile robot roda mekanum menggunakan lokalisasi dan pemetaan menggunakan lidar. Robot beroda mekanik adalah robot yang dapat bergerak dari titik A ke titik B secara mandiri. Tujuan dari penelitian ini adalah untuk mengevaluasi ketepatan dan akurasi pemetaan dan ketepatan dalam pemetaan untuk memastikan robot dapat beroperasi dengan efisien serta membangun kemampuan untuk melakukan pemetaan lingkungan sekitar secara real-time menggunakan data yang diperoleh dari sensor *Lidar* A2M12 dan mengimplementasikan algoritma *SLAM* (*Simultaneous Localization and Mapping*) untuk menentukan posisi dan orientasi robot secara simultan saat melakukan pemetaan. Untuk bergerak secara mandiri, robot harus menyadari lingkungannya dan posisinya dalam lingkungan tersebut. Metode yang digunakan adalah lokalisasi dan pemetaan simultan dengan menggunakan sensor *RPLidar* A2M12 dan *ROS* (*Robot Operating System*). Berdasarkan hasil pengujian *gmapping SLAM* sebesar 3,34%, error pengukuran jarak dan sudut sensor sebesar 1,16%, secara keseluruhan, robot otonom ini dapat digunakan bahkan di area terbuka dan rintangan sederhana.

Kata Kunci: Gerak Otonom, Lidar, Ros, Slam

Abstract

This article reviews an autonomous mobile robot with mecanum wheels using localization and mapping with Lidar. The mecanum wheeled robot is capable of autonomously moving from point A to point B. This study aims to determine the level of accuracy and precision in mapping to ensure the robot can operate efficiently, as well as to develop the ability to perform real-time environmental mapping using data obtained from the Lidar A2M12 sensor. It also aims to implement the SLAM (Simultaneous Localization and Mapping) algorithm to simultaneously determine the robot's position and orientation while mapping. For independent movement, the robot must be aware of its surroundings and its position within that environment. The method used is simultaneous localization and mapping using the RPLidar A2M12 sensor and ROS (Robot Operating System). Based on the testing results, the gmapping SLAM error rate is 3.34%, with a sensor distance and angle measurement error of 1.16%. Overall, this autonomous robot can be used even in open areas and with simple obstacles.

Keywords: LIDAR, ROS, SLAM, Autonomous Movement

1. Pendahuluan

Ilmu pengetahuan dan teknologi robotika telah berkembang pesat. Robot yang otonom dan cerdas semakin banyak diterapkan di berbagai industri. Penelitian tentang mobil tanpa pengemudi adalah salah satu bidang dengan pertumbuhan paling tinggi [1]. Teknologi di balik mobil tanpa pengemudi ini merupakan kemajuan dari robot otonom. Dengan menggunakan sistem navigasi, robot *mobile* otonom harus mampu melakukan pemetaan dan lokalisasi lingkungannya serta menentukan posisinya di dalam lingkungan tersebut [2]. Robot *mobile* otonom ini dapat digunakan untuk pekerjaan di area yang

berisiko mengalami pelanggaran keamanan. Dengan demikian, manusia tidak lagi perlu melakukan pemetaan secara manual di daerah berbahaya [3]. Tujuan pembuatan robot *mobile* ini adalah untuk mengantikan manusia dalam melakukan aktivitas pemetaan di daerah yang sulit dijangkau atau berbahaya. Salah satu teknik pemetaan yang paling dasar adalah dengan menggunakan sensor ultrasonik untuk mengukur jarak suatu objek dari hambatan di sekitarnya. Namun, sensor ultrasonik pada robot *mobile* ini memiliki beberapa kelemahan, termasuk keterbatasan dalam pembacaan jarak yang pendek dan ketepatan pengukuran sudut yang rendah [5]. Oleh

karena itu, *LIDAR* (*sensor Light Detection and Ranging*) digunakan untuk mengatasi keterbatasan pembacaan jarak dan sudut tersebut [4]. Dengan memutar sensor cahaya, teknologi *LIDAR* mengukur jarak suatu objek dengan mengukur jarak objek-objek di sekitar sensor [6]. Untuk memungkinkan robot ber navigasi secara otomatis, informasi lokasi mengenai posisi robot juga diperlukan, mengetahui peta suatu tempat saja tidak cukup. Teknik yang disebut *SLAM* (*Simultaneous Localization and Mapping*) digunakan untuk membuat peta sekaligus menunjukkan posisi robot pada peta tersebut [7]. Pendekatan *gmapping* adalah salah satu yang dapat diterapkan dalam algoritma *SLAM* [8].

G-Mapping adalah salah satu metode *SLAM* yang berbasis pada pendekatan grid. Dalam *gmapping*, lingkungan dibagi menjadi sel-sel grid kecil, dan robot menggunakan data sensor untuk menentukan keberadaan atau ketidadaan objek di setiap sel grid. Hasilnya adalah peta berbasis grid yang mendetail dari lingkungan. Penerapan *gmapping SLAM* pada autonomous mobile robot ini dijalankan pada *ROS* (*Robot Operating System*).

Middleware yang dikenal sebagai *ROS* berfungsi sebagai penghubung antara perangkat lunak dan perangkat keras robot. Tersedia banyak paket untuk *ROS* yang dapat membantu dalam penerapan metode *SLAM* [8]. Pengguna *ROS* (*Robot Operating System*) juga bertujuan untuk memudahkan para pengembang robot dalam membangun sistem robot yang diperlukan tanpa harus memulai pemrograman dari awal.

2. Metode Penelitian

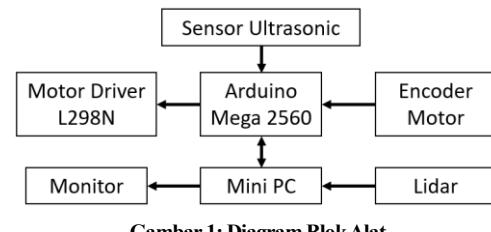
Penelitian ini menggunakan robot mecanum 4 roda. Metode *SLAM* yang digunakan adalah *g-mapping* yang memanfaatkan data jarak menggunakan sensor *LIDAR*. Algoritma *gmapping SLAM* dijalankan pada *ROS*. *ROS* bersifat *open source*, sehingga peneliti yang mempelajari robot, pengembang, dan masyarakat umum dapat menggunakan modul-modul yang disediakan oleh *ROS* secara langsung. Selain itu, pengguna juga dapat turut berkontribusi dalam pembuatan modul-modul tambahan agar *environment ROS* bisa terus berkembang.

2.1 Gambaran Umum

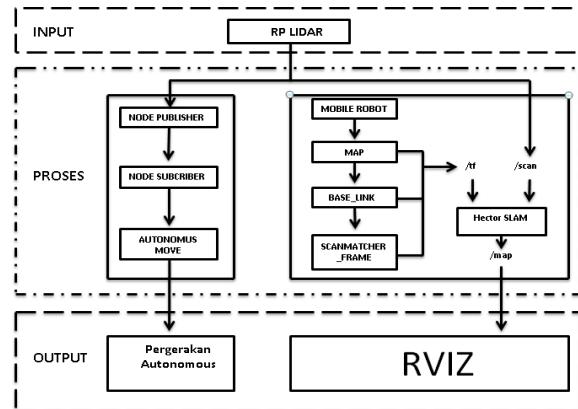
Robot *mecanum* dimaksudkan untuk memberikan representasi peta dari tata letak ruangan. Sensor yang digunakan untuk pemetaan ruang adalah *LIDAR*. Sensor ini mengukur waktu yang dibutuhkan untuk sinyal pantul diterima oleh penerima *LIDAR* setelah memproyeksikan sinar laser atau sumber sinyal lainnya [9]. Sensor *LIDAR* diakses melalui sebuah PC kecil. PC kecil ini menerima data *LIDAR* dan memprosesnya untuk membuat peta. *ROS* mengintegrasikan tampilan dan pc kecil tersebut [10].

Pengguna memiliki opsi untuk mengendalikan robot secara manual atau membiarkannya ber navigasi secara otomatis. Kontrol ini dilakukan melalui perintah yang dikirimkan dari PC kecil ke mikrokontroler.

Diagram blok peralatan pada Gambar 1 menunjukkan hardware yang digunakan dalam penelitian ini. Prosedur operasional Robot *Mobile* Otonom yang memanfaatkan pendekatan *SLAM* berbasis *LIDAR* digambarkan dalam Gambar 2, diagram *SLAM*. Jenis sensor *LIDAR* yang digunakan dalam penelitian ini adalah *RPLidar A2M12*. Model ini memiliki akurasi yang tinggi untuk sensor jarak dan sudut. Sensor *RPLidar A2M12* memungkinkan robot *mobile* untuk melaksanakan dua tugas sekaligus. Yang pertama adalah pergerakan otomatis, yang memanfaatkan data jarak dan sudut dari sensor, dan yang kedua adalah pemetaan dan lokalisasi, yang memanfaatkan teknik *SLAM* dari *Robot Operating System (ROS)*. Tujuan dari pengujian pemetaan *gmapping* adalah untuk mengevaluasi seberapa akurat hasil pemetaan dibandingkan dengan arena yang sebenarnya. Pemetaan ini dibuat berdasarkan pengukuran sudut dan jarak dari sensor *RPLidar A2M12*, yang diproses oleh *gmapping* dan diterapkan rumus yang diperlukan untuk menentukan rata-rata kesalahan.



Gambar 1: Diagram Blok Alat



Gambar 2: Diagram *SLAM*

2.2 Robot Mecanum

Pada penelitian ini menggunakan robot mecanum 4 roda seperti pada gambar 3. Design robot ini mengambil inspirasi dari robot mecanum NEXUS yang ada di pasaran. Berdasarkan design dari robot tersebut, kemudian dikembangkan design robot ini dengan berbagai penyesuaian dengan *hardware* yang digunakan pada penelitian ini. Tabel 1

memperlihatkan hardware yang digunakan, yaitu menggunakan mikrokontroller Arduino Atmega 2560, dan motor DC 12V dengan encoder beserta driver motor L298N sebagai penggerak robot. Robot ini beroperasi 1,5 jam, dengan ukuran robot yaitu: 379 mm x 210 mm x 90 mm diameter roda yaitu 97 mm dan berat robot 3.2 kg



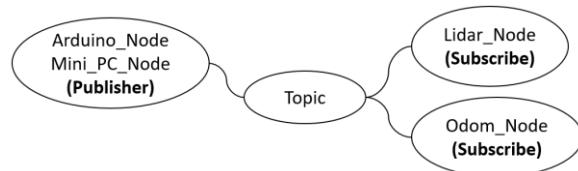
Gambar 3: Robot Mecanum 4 Roda

Tabel 1. Spesifikasi Robot Mobile

Parameter	Spesifikasi
Berat	3.2 kg
Ukuran (p x l x t)	379mm x 210mm x 90mm
Waktu isi baterai	2 jam
Waktu operasi	1,5 jam
Seri lidar	A2M12
Mikrokontroler	Arduino Atmega 2560
Motor driver	L298N

2.3 ROS

Framework *open-source* yang disebut *Robot Operating System (ROS)* berisi pustaka dan alat yang berguna dalam membuat aplikasi atau program untuk sistem robot [11]. Karena desainnya yang modular, *ROS* memudahkan integrasi antara program yang baru dibangun dengan yang sudah ada. Program-program, yang disebut node, saling berkomunikasi melalui topik untuk memfasilitasi komunikasi antar proses dalam *ROS*. *Node* adalah perangkat lunak yang dirancang untuk melaksanakan berbagai aktivitas atau tugas. Sebaliknya, topik berfungsi sebagai saluran yang digunakan node untuk berkomunikasi satu sama lain. Setiap topik memiliki tipe pesan dengan format tertentu [12].



Gambar 4: Komunikasi Antar Node

Dua jenis node yang berbeda pada Gambar 4 adalah *publisher* dan *subscriber*. Sebuah node umumnya tidak mengetahui dengan *node* mana ia berinteraksi. Namun, node dapat mempublikasikan data ke topik tertentu. *Arduino_node* dan *mini_pc_node* berfungsi sebagai *publisher*, mengirimkan data ke topik untuk memungkinkan komunikasi antar *node*. Selanjutnya, data tersebut dikirimkan ke *subscriber lidar_node* dan *odom_node*. Data diterima oleh *lidar_node* untuk membaca peta, dan oleh *odom_node* untuk menentukan posisi robot.

2.4 Algoritma SLAM

SLAM adalah teknik komputasi yang mencatat jejak robot dan membuat peta lingkungan sekitarnya, serta memperbarui peta tersebut berdasarkan data sensor [13]. Algoritma *SLAM* dapat diimplementasikan menggunakan berbagai teknik, seperti *hector_slam*, *gmapping*, dan *cartographer*. Semua algoritma *SLAM* ini termasuk dalam paket standar *ROS* [14]. Prosedur pemetaan dan lokalisasi dalam pekerjaan ini dilakukan oleh penulis menggunakan *SLAM gmapping*. Teknik *SLAM* lainnya, termasuk *hector_slam*, dapat dibandingkan dengan metode *SLAM gmapping*. Karena menggunakan algoritma open-loop dan dapat menghasilkan peta meskipun tidak ada data odometri robot. Akibatnya, selama proses pemetaan, kesalahan yang dihasilkan dari ketidakpastian sensor akan meningkat. Sebaliknya, *gmapping* menggunakan metodologi closed-loop, di mana peta yang dihasilkan akan diperbarui jika robot kembali ke posisi awalnya [15]. Proses pemetaan dan pelokalan di *SLAM* dilakukan secara bersamaan dimana mobile robot secara relatif membuat peta. Peta yang dibuat digunakan untuk menghitung dan memperkirakan posisi landmark dan lintasan *mobile robot* [16].

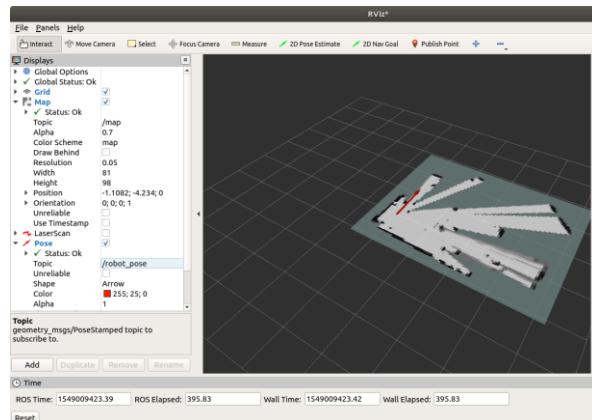
2.5 Gmapping SLAM

Gmapping merupakan salah satu algoritma *SLAM* berbasis *particle filter* atau juga dikenal dengan Monte Carlo Localization (MCL) yang banyak digunakan dalam robotika untuk pembuatan peta lingkungan secara real-time. *GMapping* memanfaatkan algoritma loop closure dan mengandalkan data odometri dalam proses pembentukan peta lingkungan. Teknik loop closure yang digunakan memungkinkan sistem memperbaiki atau memperbarui peta saat robot kembali ke posisi semula. Algoritma ini dikenal lebih efisien dan membutuhkan daya komputasi yang lebih rendah dibandingkan dengan metode lainnya, sehingga cocok digunakan untuk keperluan keamanan. Saat membuat peta, *GMapping* menggunakan metode peta grid okupansi (*occupancy grid map*) dalam bentuk dua dimensi (2D). *Gmapping* menjadi populer karena implementasinya yang efisien dalam *ROS*.

(Robot Operating System) dan kemampuannya bekerja pada sistem komputasi dengan sumber daya terbatas [17].

2.6 RViz

Sistem ROS menggunakan *RViz*, yang merupakan kependekan dari Robot Visualization. *RViz* merupakan alat untuk menampilkan data sensor dan informasi dari robot secara real time dalam bentuk visual 3D. Dengan menggunakan informasi dari sensor seperti *LIDAR*, kamera, dan odometri, serta keluaran algoritma seperti *SLAM* atau perencanaan jalur, pengguna dapat melihat bagaimana robot "melihat" lingkungannya dengan *RViz*. Gambar 5 berikut ini merupakan tampilan dari software *RViz* yang sedang menampilkan visualisasi data *LIDAR* pada suatu lapangan.



Gambar 5: Tampilan Software *RViz*

2.7 Mean Absolute Persentage Error (MAPE) dan Mean Absolute Error (MAE)

MAPE adalah metrik evaluasi yang digunakan untuk mengukur akurasi model hasil pengukuran dalam hal persentase. Presentase ini memberikan gambaran tentang sejauh mana kesalahan pengukuran berbanding dengan nilai aktual dalam bentuk persentase.

$$MAPE = \left| \frac{y_i - \hat{y}_i}{y_i} \right| \cdot 100\% \quad (1)$$

Keterangan:

y_i = nilai hasil aktual

\hat{y}_i = nilai hasil pada *RViz*

Rata-rata error sering kali diukur menggunakan *Mean Absolute Error (MAE)* konteks prediksi. *MAE* memberikan gambaran tentang seberapa besar kesalahan rata-rata antara hasil pengukuran dan nilai aktual.

$$MAE = \frac{1}{n} \sum_{i=1}^n |X| \quad (2)$$

Keterangan:

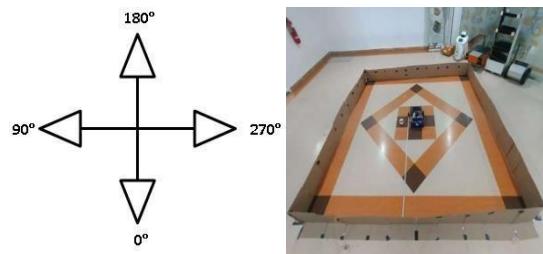
n = jumlah data

X = nilai hasil persentase error dari *MAPE*

3. Hasil dan Pembahasan

3.1 Pengujian Sensor RPLidar A2M12

Pengujian melibatkan penggunaan *LIDAR* untuk membuat peta dengan area berbentuk persegi berukuran 330 cm x 330 cm seperti terlihat pada gambar 6. Namun, karena algoritma yang digunakan untuk pergerakan otomatis hanya memungkinkan gerakan dalam arah *vertical* dan *horizontal*, robot tidak dapat mengikuti dinding ruangan. Bentuk ruangan yang dihasilkan dalam pengujian ini sesuai dengan bentuk ruangan yang sebenarnya. Tujuan dari pengujian sensor ini adalah untuk memastikan seberapa akurat pembacaan jarak yang sebenarnya dibandingkan dengan jarak yang diukur oleh sensor. Sensor *RPLidar A2M12* dengan jangkauan deteksi 360° dapat mengukur jarak dan sudut. Pengujian jarak ini dilakukan dalam 5 kali percobaan, dengan sampel sudut 0°, 90°, 180°, dan 270°.



Gambar 6: Arena Uji

Coba

Tabel 2 Hasil pengujian sensor *RP Lidar*

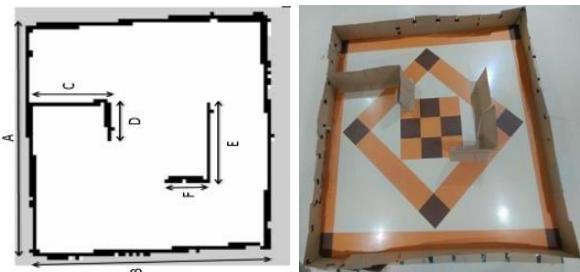
No	Ukuran Hasil ROS (cm)				MAPE (%)			
	0°	90°	180°	270°	0°	90°	180°	270°
1	168,6	164	164,4	176	1,56	0	0,85	2,32
2	168,6	164	164,4	176	1,56	0	0,85	2,32
3	168,6	164	164,2	176	1,56	0	0,85	2,32
4	168,6	164	164,2	176	1,56	0	0,85	2,32
5	168,6	164	164,2	176	1,56	0	0,85	2,32
MAE (%)				1,16%				

Pengujian *lidar* ini bertujuan untuk mengetahui keakuratan *lidar*. Nilai ukuran yang ditampilkan di *RViz*, dengan cara membandingkan hasil ukuran arena sebenarnya. Hasil pengukuran pada map yang telah dibuat dari pengujian tabel diatas ukuran hasil map aktual dari 1 sampai 5 yaitu 0°=166, 90° =164, 180° =163, dan 270 ° =172. Gambar 6 arena percobaan pengujian lidar dan dari hasil tabel 2 menghasilkan rata-rata error MAE keseluruhannya sebesar 1,16%

3.2 Pengujian Mapping Pada GMapping SLAM

Tujuan dari pengujian pemetaan dengan gmapping adalah untuk menilai seberapa akurat hasil

pemetaan dibandingkan dengan arena yang sebenarnya. Pemetaan ini didasarkan pada pengukuran sudut dan jarak dari sensor *RPLidar A2M12*, yang telah dianalisis oleh *gmapping* dan menghitung rata rata error. Pengujian dilakukan dengan menggunakan ruangan buatan yang memiliki ukuran 330cm x 330cm. Robot dijalankan untuk menelusuri arena yang ada. Data *LIDAR* dikirim ke laptop untuk kemudian diolah dan ditampilkan dalam tools *RViz*. Semua proses data yang dilakukan selama pemetaan ruangan ini disimpan dalam *rosbag*, yang kemudian dilakukan analisa terhadap peta yang telah dibentuk.



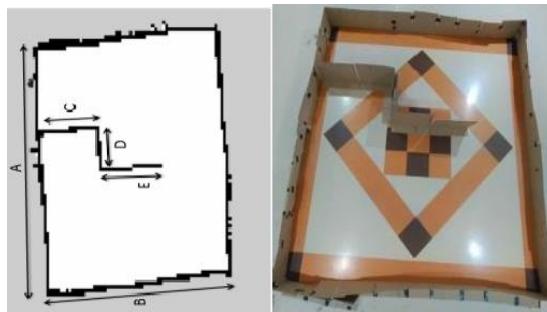
Gambar 7: Desain Ruangan Percobaan 1

Gambar 7 merupakan hasil percobaan pertama sebelah kiri gambar hasil pemetaan di hasilkan dari *lidar* ditampilkan di *RViz* dan gambar sebelah kanan map yang sebenarnya.

Tabel 3 Hasil pengukuran pemetaan percobaan 1

Indeks dinding	Ukuran Aktual (cm)	Ukuran Pada <i>RVIZ</i> (cm)	MAPE (%)
A	330	352,1	1,48
B	330	329,2	0,24
C	109	103	5,50
D	55	57	3,63
E	110	110	0
MAE (%)		2,17	

Tabel 3 menunjukkan ada 5 ukuran aktual dan ukuran pada *RViz*. Ukuran tersebut di hitung *MAPE* nya satu per satu dari indeks dinding A sampai E. Dan merupakan hasil analisis pada percobaan ke satu. Mendapatkan *MAE* 2,17 % kemudian lanjut ke percobaan kedua pada gambar 8.



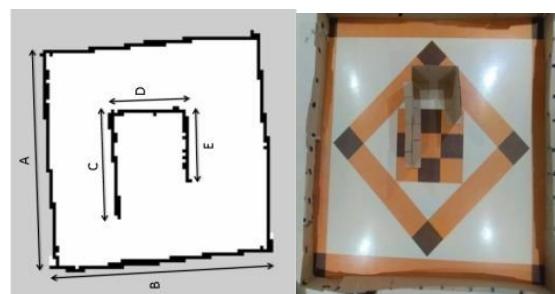
Gambar 8: Desain Ruangan Percobaan 2

Gambar 8 merupakan hasil percobaan kedua yang mana sebelah kiri adalah gambar hasil pemetaan dihasilkan dari *lidar* ditampilkan di *RViz*. Sedangkan gambar sebelah kanan map yang sebenarnya.

Tabel 4 menunjukkan ada 5 ukuran aktual dan ukuran pada *RViz*, ukuran tersebut di hitung *MAPE* nya satu per satu dari indeks dinding A sampai E sehingga hasil analisis pada percobaan ke dua mendapatkan *MAE* 2,93 % kemudian lanjut ke percobaan kedua pada gambar 9.

Tabel 4 Hasil pengukuran pemetaan percobaan 2

Indeks dinding	Ukuran Aktual (cm)	Ukuran Pada <i>RVIZ</i> (cm)	MAPE (%)
A	329	326	0,91
B	330	331,1	0,33
C	110	113	2,72
D	55	55	0
E	108	115	2,77
F	55	61	10,90
MAE (%)		2,93	



Gambar 9: Desain Ruangan Percobaan 3

Gambar 9 hasil percobaan ketiga, sebelah kiri gambar hasil pemetaan dari *RViz* dan gambar sebelah kanan map yang sebenarnya.

Tabel 5 Hasil pengukuran pemetaan percobaan 3

Indeks dinding	Ukuran Aktual (cm)	Ukuran Pada <i>RVIZ</i> (cm)	MAPE (%)
A	329	321,8	2,18
B	330	329,4	0,18
C	164	157,9	3,71
D	110	127,2	15,63
E	110	106,8	2,90
MAE (%)		4,92	

Tabel 5 menunjukkan ada 5 ukuran aktual dan ukuran pada *RViz*. Ukuran tersebut di hitung *MAPE* nya satu per satu dari indeks dinding A sampai E sehingga hasil analisis pada percobaan ke tiga mendapatkan *MAE* 4,92 %. Percobaan ketiga adalah yang terakhir. Dari hasil percobaan kesatu sampai percobaan ketiga semua hasil *MAE* berbeda sehingga hasil analisi

keseluruhan ketiga percobaan tersebut yaitu 3,34%.

4. Kesimpulan

Dari hasil penelitian implementasi pemetaan robot berbasis *lidar* disimpulkan bahwa *lidar* a2m12 masih terdapat error dan analisis yang telah didapatkan. Dari hasil pengujian jarak dan sudut dengan sampel sudut tengah ke 0° , 90° , 180° , dan 270° pada sensor *Lidar* A2M12. Hasil percobaan 1 sampai 5 memiliki *MAE* sebesar 1,16 %. Untuk pengujian *mapping* terhadap jarak sebenarnya dari hasil menggunakan *lidar* terhadap percobaan ke satu yaitu *MAE* 2,17%. Percobaan kedua memiliki *MAE* 2,93% dan yang terakhir percobaan ke tiga memiliki *MAE* 4,92%. Keseluruhan arena hasil pengujian *gmapping slam* memiliki *MAE* sebesar 3,34%. Dari pengujian jarak dan sudut *lidar* memiliki *MAE* sedikit dan pengujian *gmapping slam* *MAE* naik sedikit dari pengujian sudut dan jarak. Untuk pengembangan selanjutnya agar lebih akurat mencoba untuk memakai *lidar* yang tipe lebih bagus.

Ucapan Terimakasih

Penulis mengucapkan terimakasih kepada Politeknik Negeri Batam karena telah menyediakan tempat dan juga mendanai penelitian ini. Penelitian ini dilaksanakan dalam rangka penerapan pembelajaran berbasis proyek (PBL).

Daftar Pustaka

- [1] Wolcott, Ryan W., and Ryan M. Eustice. "Visual localization within lidar maps for automated urban driving." *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014
- [2] Rahman, Aulia. "Penerapan *slam gmapping* dengan *robot operating system* menggunakan *laser scanner* pada *turtlebot*." *Jurnal Rekayasa Elektrika* 16.2 (2020).
- [3] Prayoga, S. et al. "EKF SLAM menggunakan *lidar*" (2010).
- [4] Nafis, Zishwa Muhammad Jauhar. "Pemetaan Ruangan Berbasis LIDAR Menggunakan Mobile Robot." (2019).
- [5] Zahra, Lilik, Muhammad Ikhsan Sani, and Simon Siregar. "Perancangan Dan Implementasi Mapping System Untuk Navigasi Roner (Robot Cleaner)." *eProceedings of Applied Science* 4.3 (2018).
- [6] Fikri, Muhammad, and Muhammad Rivai. "Sistem Penghindar Halangan Dengan Metode LIDAR Pada Unmanned Surface Vehicle." *Jurnal Teknik ITS* (SINTA: 4, IF: 1.1815) 8.2 (2020).
- [7] Alsadik, Bashar, and Samer Karam. "The simultaneous localization and mapping (*SLAM*)-An overview." *Journal of Applied Science and Technology Trends* 2.02 (2021).
- [8] Taufik, Ahmad Sulkhan. Sistem navigasi *waypoint* pada *autonomous mobile robot*. Diss. Brawijaya University, 2013.
- [9] Corke, Peter, and Vision Robotics. "Control Fundamental algorithms in MATLAB." *Springer Tracts in Advanced Robotics*, (Springer Berlin Heidelberg, 2011) (2017).
- [10] Saat, Shahrizal, et al. "Hector slam 2d mapping for simultaneous localization and mapping (*slam*)."*Journal of Physics: Conference Series*. Vol. 1529. No. 4. IOP Publishing, 2020.
- [11] Quigley, Morgan, et al. "ROS: an open-source Robot Operating System." *ICRA workshop on open source software*. Vol. 3. No. 3.2. 2009.
- [12] Pyo, Y., et al. "ROS Robot Programming (English). ROBOTIS, 12 2017
- [13] Leonard, John J., and Hugh F. Durrant-Whyte. "Simultaneous map building and localization for an autonomous mobile robot." *IROS*. Vol. 3. 2019.
- [14] Fx, Dieter, et al. "A probabilistic approach to collaborative multi-robot localization." *Autonomous robots* 8 (2000).
- [15] Grisetti, Giorgio, Cyrill Stachniss, and Wolfram Burgard. "Improved techniques for grid mapping with rao-blackwellized particle filters." *IEEE transactions on Robotics* 23.1 (2007).
- [16] Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (SLAM): Part I," *IEEE Robotics & Automation Magazine*, vol. 13, pp. 99-110, 2006
- [17] Firmansyah, R. A., Prabowo, Y. A., Suheta, T. ., & Utomo, A. N. D. (2024). *Implementation of SLAM Gmapping and Extended Kalman Filter for Security Robot Navigation System*. Emitor: *Jurnal Teknik Elektro*, 24(2), 145–153.